УДК 004.318

А.В. Палагин, Н.Г. Петренко

Методологические основы разработки лингвистического процессора для обработки лингвистических корпусов текстов сверхбольших объемов. II*

Разработаны методологические основы построения быстродействующих аппаратных лингвистических процессоров. Рассмотрена практическая реализация аппаратных морфологических процессоров, позволяющих на два и более порядка повысить производительность лингвистического анализа корпусов текстов большого объема в сравнении с программной реализацией.

The methodological basis for the construction of the high-speed hardware linguistic processors is elaborated. The practical implementation of the morphological hardware processors which enable to improve the performance of the linguistic analysis of large volume text corpus two times more comparing with the software implementation is considered.

Розроблено методологічні основи побудови швидкодіючих апаратних лінгвістичних процесорів. Розглянуто практичну реалізацію апаратних морфологічних процесорів, які дозволяють на два і більше порядки підвищити продуктивність лінгвістичного аналізу корпусів текстів великого обсягу порівняно з програмною реалізацією.

Введение. Описаны методологические основы построения аппаратных лингвистических процессоров (АЛП) для обработки корпусов текстов (в научно-технической сфере) большого объема, в частности аппаратных морфологических процессоров (АМП).

Постановка задачи

Одной из важных задач разработки общей теории компьютерной обработки предметных знаний, представленных в естественно-языковой (ЕЯ) форме, считается построение эффективных лингвистических процессоров (ЛП). Эта задача особенно актуальна для приложений обработки лингвистических корпусов текстов (ЛКТ) сверхбольших объемов (и в реальном времени).

Поэтому задача существенного (на два порядка и более) повышения быстродействия лингвистического анализа актуальна. Следует отметить, что такое повышение быстродействия может быть достигнуто за счет дополнительных аппаратурных затрат как стандартной, так и специализированной разработки. Аппаратные средства (АС) первого типа – это продукты известных фирм, доступные на рынке и прилагаемая к ним система автоматизированного проектирования (САПР). Несомненным лидером таких АС на рынке есть платы с установленными на них программируемыми логическими интегральными схемами (ПЛИС), в которых есть сверхбыстродействующая память и быстродействующая память большого объема [1]. АС второго типа – специализированная разработка, для которых необходимо спроектировать архитектурно-структурную организацию процессора, электрическую схему или граф-схемы алгоритмов, специальное программное обеспечение управления ими и драйверы совмещения с операционной системой компьютера. При реализации лингвистического процессора оба эти варианта АС имеют свои преимущества и недостатки. Для АС первого типа к преимуществам относится их доступность на рынке, их вычислительная мощность постоянно увеличивается разработчиками, к ним уже прилагается программное обеспечение, а проект АЛП может быть разработан за время от двух месяцев. Недостаток этих АС – низкий процент использования установленного на плате оборудования. К преимуществам АС второго типа следует отнести повышение быстродействия на один–два порядка в сравнении с АС первого типа, что служит главным критерием при разработке АЛП. А к недостаткам – необходимость коллектива разработчиков (системотехников и программистов) и время разработки проекта – от одного года.

*Продолжение. Начало см. в № 2, 2014 нашего журнала.

Повышение быстродействия реализации алгоритма лингвистического анализа для обоих типов АС достигается путем перевода операторов алгоритмического и программного уровней (реализация лингвистического анализа программным способом) на нижние уровни интерпретации [2]: для АС первого типа – на микропрограммный уровень, для АС второго типа – на микропрограммный и частично на физический уровни.

В [2] приведены дополнительные доводы целесообразности реализации ЛП в целом, и морфологического процессора в частности аппаратными средствами. Например, аппаратная реализация дает возможность параллельной обработки всех слов одного предложения одновременно. При этом упрощаются алгоритмы синтаксического и семантического анализа.

Методологические основы разработки АЛП в статье представлены следующими компонентами[1]:

- онтологический подход к построению аппаратных средств лингвистического анализа естественно-языковых объектов (ЕЯО);
- разработка функциональной схемы АЛП;
- разработка подсистем АЛП;
- задача оптимального синтеза АЛП;
- структурная организация и проектирование АМП;
- структурная организация АМП для обработки ЛКТ разного объема;
- оценки сложности структурной реализации АМП.

Структурная организация и проектирование аппаратных морфологических процессоров

Описана аппаратная реализация подсистемы морфологического анализа (или аппаратного морфологического процессора), причем только последовательного анализа словоформ входного предложения. Как указано в [3], для реализации параллельной обработки всех словоформ предложения потребуется K блоков морфологического анализа, где K – максимальное количество вхождений словоформ в предложение.

Общая схема реализации морфологического анализа (МА), независимо от способа реализации, сводится к приему последовательности слов, составляющих входной текст, распознаванию или дешифрации анализируемого слова и нахождения соответствующей ему «точки в гиперпространстве» (или реализация табличного метода анализа), в которой анализируемому слову приписаны все необходимые морфологические характеристики. Это «гиперпространство» представляет собой по осям X_i части речи заданного ЕЯ, где $i = \overline{1,n}$, n – количество частей речи, а по осям Y_i – последовательность словоформ i-й части речи.

Описанная ранее последовательность шагов МА «идеальна» и практически нереализуема для современных микроэлектронных технологий, а приближение к ней возможно только для аппаратной реализации алгоритма МА. Для «идеальной» реализации понадобился бы дешифратор (или память) с адресацией 2^{256} разрядов. Этот параметр определен из того, что для кодирования одной буквы (символа) слова требуется 8 бит (при однобайтовом кодировании символов), а максимальное количество символов самых длинных словоформ в ЛБД общеупотребительной лексики украинского языка «Словники України» равно 32. Отсюда и получена степень двойки ($8 \times 32 = 256$).

Классический программный МА выполняется последовательно по буквам, начиная с окончания, нахождения основы словоформы и формирования последовательности омонимов анализируемого слова. При этом для каждого омонима формируется свое множество морфологических характеристик.

Отметим, что в общем случае только «идеальная» аппаратная реализация позволяет избежать раздельного анализа окончания и основы словоформы.

Таким образом, если условно расположить на плоскости по оси X реализацию МА классическим программным способом, то описанная «идеальная» аппаратная реализация будет рас-

[1] Первые четыре компоненты рассмотрены в [3]. В данной статье будут рассмотрены компоненты 5–7.

положена по оси *Y*, а все другие реализации будут расположены между ними и будут множеством Парето решений реализации АМП (рис. 1).

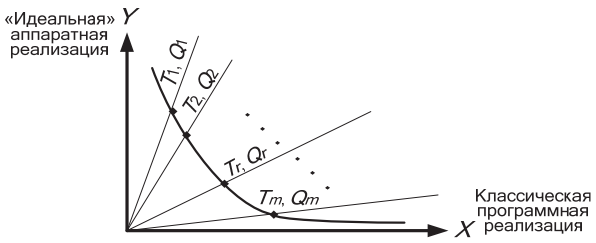

Рис. 1. Множество Парето решений реализации морфологического процессора

Обобщенная схема АС реализации алгоритма МА для некоторого решения T_r, Q_r представлена на рис. 2, где приняты следующие обозначения:

m – количество слов в анализируемом тексте. Эта последовательность формируется на этапе графематического анализа, записывается в память слов текста и есть исходными данными для МА;

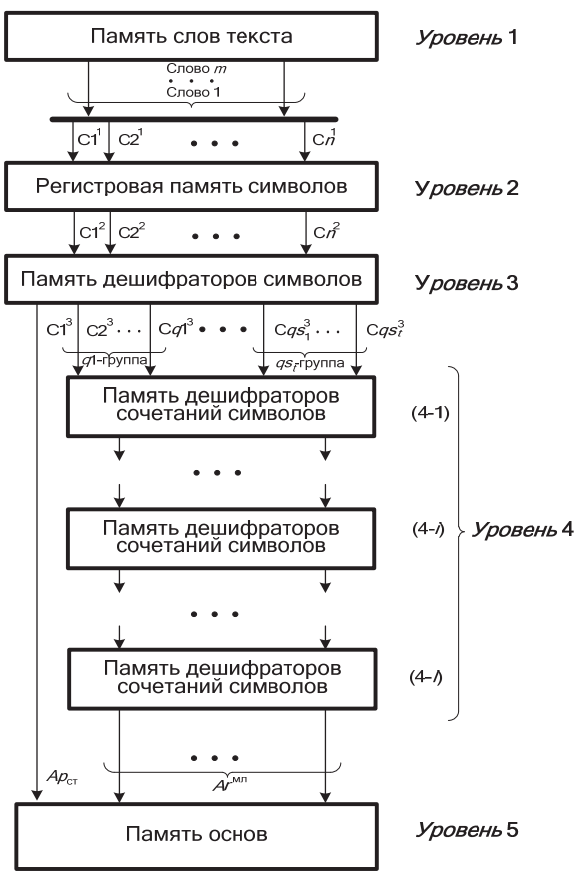

Рис. 2. Обобщенная схема аппаратных средств МА

$C_1^1, C_2^1, ..., C_n^1$ – максимальное количество букв (символов) в словах анализируемых текстов;

$C_1^3, C_2^3, .., C_{q1}^3, ..., C_{qs_1}^3, ..., C_{qs_t}^3$ – первая буква слова и *qs* групп сочетаний символов (начиная со второго), которые формируются на основе статистических характеристик и заданных ограничений на оборудование;

$A_p^{ст}$ – старшие разряды адреса памяти слов;

$A_r^{мл}$ – младшие разряды адреса памяти слов.

Суть построения схемы заключается в «усечении» адресного пространства, необходимого для «идеальной» реализации, до адресного пространства памяти, представленной на стандартном оборудовании. Для этого служит уровень 4 (рис. 2).

Структурная организация аппаратных морфологических процессоров для обработки лингвистических корпусов текстов разного объема

Структурная организация АС морфологического анализа естественно-языковых текстов (ЕЯТ), составляющих некоторый ЛКТ, и затраты оборудования сильно зависят от статистических характеристик заданного корпуса текстов, в частности от количества употребляемых словоформ *K*, их средней длины $L_{ср}$, количества сочетаний символов (начиная со второго), перекрывающих среднюю длину *K*, и ряда других.

Диаграмма зависимости количества основ от их длины (количества символов в основе) для общеупотребительной лексики украинского языка приведена на рис. 3. Кривая на диаграмме наиболее близко аппроксимируется экспоненциальной функцией вида $f(x) = 33600\,e^{-\frac{(x-9)^2}{16}}$ с величиной достоверности аппроксимации $R2 = 0{,}982$. Вычисление величины R^2 в *Microsoft Excel* показано ниже:

$$R^2 = 1 - \frac{SSE}{SST},$$

где $SSE = \sum\left(Y_j - \hat{Y}_j\right)^2$ и $SST = \left(\sum Y_j^2\right) - \frac{\left(\sum Y_j\right)^2}{n}$.

Суть задачи построения АМП сводится к сокращению аппаратного оборудования *Qp* (рас-

смотрим только стандартное оборудование на платах с ПЛИС, так как разработка специального оборудования представляет собой самостоятельную научно-техническую проблему). Исследования показали, что достаточно обеспечить независимую адресацию сочетаний символов, перекрывающих среднюю длину $L_{ср}$ словоформ (или основ) заданного ЛКТ.

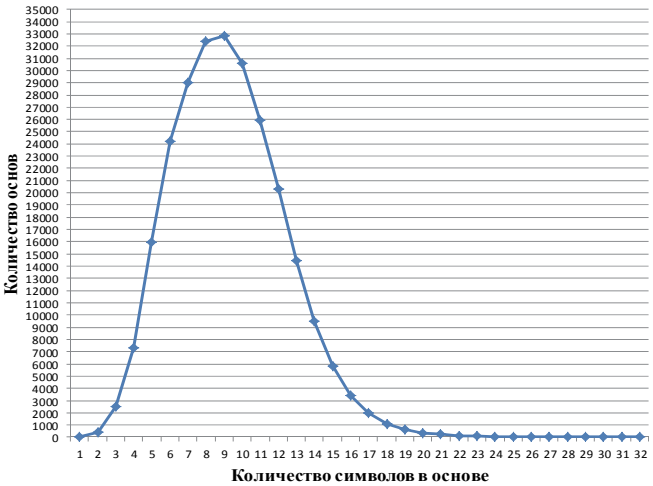

Рис. 3. Диаграмма зависимости количества основ от их длины

Например, для ЛКТ общеупотребительной лексики украинского языка (все основы представлены в ЛБД «Словники України»), в котором средняя длина слова составляет 9,27 символа, необходимо обеспечить перекрытие до десятого символа и более. Также следует учесть необходимость адресации (до четырех разрядов) различных форм глагола, имеющих одинаковые последовательности символов и перекрывающие $L_{ср}$. Адресацию остальных сочетаний символов можно «собрать» по схеме ИЛИ. Статистические характеристики для произвольного ЛКТ вычисляются в приложении *Microsoft Excel* стандартными функциями после преобразования текста в таблицу в *Microsoft Word*.

Далее будет рассмотрена разработка АМП для трех вариантов ЛКТ:

- вариант *A* – ЛКТ, содержащий общеупотребительную лексику украинского языка (табл. 1);
- вариант *B* – ЛКТ по онтологическому инжинирингу (табл. 2);
- вариант *C* – ЛКТ по онтолого-управляемым информационным системам общего назначения (табл. 3).

Различные типы плат с ПЛИС, их описание и технические характеристики представлены на веб-сайтах *www.hitechglobal.com/boards/allboards.htm* и *www.hilinx.com/products/boards_kits.htm*. На рис. 4 показана блок-диаграмма платы *HTG-V4PCIE*. На этой плате проводилось моделирование АМП для варианта *C*.

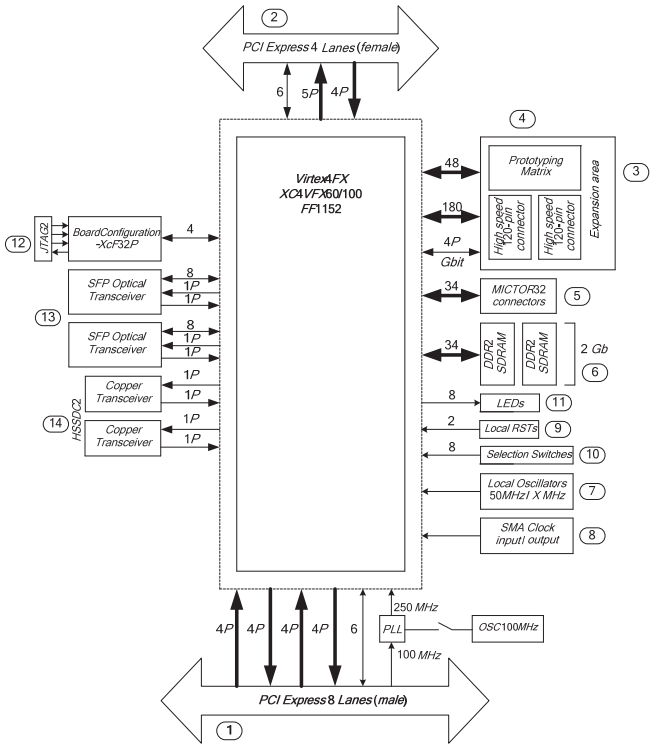

Рис. 4. Блок-диаграмма платы *HTG-V4PCIE*

Моделирование АМП для варианта *C* выполнено в системе САПР ПЛИС *Xilinx ISE* 8.2*i* с использованием платы, на которой установлены следующие аппаратные средства, доступные для пользователя и необходимые, в частности для практической реализации АМП:

- кристалл ПЛИС *Virtex*-4, содержащий 376 блоков СОЗУ 18*Kb*×1, с возможностью организации от 16*Kb*×1 до 512×36 бит (*www.hilinx.com/products/boards_kits/virtex*6.*htm*);
- внешняя (по отношению к кристаллу ПЛИС) память *RAM* – два независимых блока 64*M*×16 бит, на одном из которых реализована память основ, а на втором – память окончаний.

Количественные показатели сочетаний символов для основ ЛКТ

Таблица 1. Вариант *A*. Общее количество основ – 259209, средняя длина основы – 9,27

C2	**C3**	**C4**	**C5**	**C6**	**C7**	**C8**	**C9**	**C10**	**C11**	**C32**
35	38	37	36	6	6	35	35	35	34	1
C1–C9	**C1–C10**	**C2–C3**	**C2–C4**	**C2–C5**	**C2–C6**	**C2–C7**	**C2–C8**	**C2–C9**	**C2–C10**	**C2–C32**
206647	224417	789	26450	32208	76421	119831	152789	176251	193052	223140
C3–C32	**C4–C5**	**C4–C32**	**C5–C6**	**C5–C7**	**C5–C8**	**C5–C9**	**C5–C10**	**C5–C11**	**C5–C16**	**C5–C32**
179001	934	125805	908	7950	27987	48550	61356	69450	83290	83961
C6–C7	**C6–C8**	**C6–C9**	**C7–C10**	**C7–C11**	**C7–C32**	**C8–C9**	**C8–C32**	**C9–C11**	**C10–C11**	**C8–C10**
835	6796	21543	16290	23929	35226	759	22790	4074	649	4989
C11–C12	**C11–C13**	**C11–C16**	**C11–C32**	**C13–C14**	**C14–C16**	**C14–C32**	**C15–C16**	**C17–C18**	**C17–C19**	**C17–C32**
594	2733	6125	6639	501	1364	2156	388	293	581	703
C19–C20	**C20–C22**	**C21–C22**	**C23–C24**	**C23–C25**	**C25–C26**	**C26–C28**	**C27–C28**	**C29–C30**	**C29–C31**	**C31–C32**
198	206	110	66	72	28	17	14	6	6	2

Таблица 2. Вариант *B*. Общее количество слов – 32055, средняя длина слова – 8,54

C2	**C3**	**C4**	**C5**	**C6**	**C7**	**C8**	**C9**	**C10**	**C11**	**C31**
38	40	40	36	38	36	35	38	39	38	1
C1–C9	**C1–C10**	**C2–C3**	**C2–C4**	**C2–C5**	**C2–C6**	**C2–C7**	**C2–C8**	**C2–C9**	**C2–C10**	**C2–C31**
25301	27666	548	3135	8095	12656	16655	20293	23278	25615	29221
C3– C32	**C4–C*5***	**C4–C31**	**C5–C6**	**C5–C7**	**C5–C8**	**C5–C9**	**C5–C10**	**C5–C11**	**C5–C16**	**C5–C31**
25966	674	20909	624	3236	7016	9902	12078	13635	15542	15674
***C*6–C7**	**C6–C8**	**C6–C9**	**C7–C10**	**C7–*C*11**	**C7–C31**	**C8–C9**	**C8–C31**	**C9–C11**	**C10–C11**	
568	2736	5469	4130	5385	7306	499	4820	1362	380	
C11–C12	***C*11–C13**	**C11–C16**	**C11–C32**	**C13–C14**	**C14–C16**	**C15–C16**	**C17–C18**	**C17–C19**	**C17–C31**	
332	771	1233	1380	236	334	162	103	144	200	
C19–C20	**C20–C22**	**C21–C22**	**C23–C24**	**C23–C25**	**C25–C26**	**C26–C28**	**C27–C28**	**C29–C30**	**C29–C31**	
69	68	41	25	27	11	7	4	2	2	

Таблица 3. Вариант *C*. Общее количество слов – 9406, средняя длина слова – 9,35

C2	**C3**	**C4**	**C5**	**C6**	**C7**	**C8**	**C9**	**C10**	**C11**	**C31**
38	40	40	36	38	35	34	38	39	37	1
C1–C9	**C1–C10**	**C2–C3**	**C2–C4**	**C2–C5**	**C2–C6**	**C2–C7**	**C2–C8**	**C2–C9**	**C2–C10**	**C2–C31**
7203	7980	404	1461	2648	3653	4767	5930	6969	7748	9172
C3– C32	**C4–C*5***	**C4–C31**	**C5–C6**	**C5–C7**	**C5–C8**	**C5–C9**	**C5–C10**	**C5–C11**	**C5–C16**	**C5–C31**
8707	479	7696	463	1699	3024	4079	4862	5443	6187	6304
***C*6–C7**	**C6–C8**	**C6–C9**	**C7–C10**	**C7–C11**	**C7–C31**	**C8–C9**	**C8–C31**	**C9–C11**	**C10–C11**	
434	1468	2505	1905	2426	3301	356	2324	735	270	
C11–C12	**C11–C13**	**C11–C16**	**C11–C32**	**C13–C14**	**C14–C16**	**C15–C16**	**C17–C18**	**C17–C19**	**C17–C31**	
236	451	696	823	183	242	136	92	121	170	
C19–C20	**C20–C22**	**C21–C22**	**C23–C24**	**C23–C25**	**C25–C26**	**C26–C28**	**C27–C28**	**C29–C30**	**C29–C31**	
63	64	40	24	26	10	7	4	2	2	

Структура варианта А

Как было показано, исходными данными для проектирования АМП приняты заданные статистические характеристики ЛКТ и аппаратное оборудование.

При морфологическом анализе последовательности слов, составляющих некоторый ЕЯТ, первый символ каждого слова (а им может быть только буква) анализируется отдельно, поскольку по его значению определяется ряд грамматических показателей, таких как: первое слово предложения, аббревиатура, соотнесение слова с заданным ЕЯ и ряд других. Для украинского языка первыми буквами слов могут быть 30 букв, следовательно для их адресации необходимо пять разрядов (строчная и заглавная буквы считаются одной буквой, а их различение определяется в отдельной микропрограмме). Анализ первой буквы слова делит основную память слов на 32 сегмента, при этом 30 сегментов отводится под индексы соответствующих букв, а два сегмента свободны, и их объем достаточен для хранения всех морфологических характеристик (результатов вычисле-

ния) слов заданного ЛКТ. В «точке гиперпространства» некоторого слова хранится индекс-ссылка на соответствующий адрес в сегменте результатов. Поэтому разрядность данных памяти слов может быть сравнительно небольшой – 16–20 разрядов.

Далее необходимо выбрать сочетания символов для уровня 4 дешифраторов сочетаний символов (рис. 2). Они выбираются исходя из аппаратного оборудования, установленного на заданной плате ПЛИС, в частности зависят от объемов внутренней сверхбыстродействующей *RAM* (СОЗУ) и внешней *RAM*. Понятно, что чем больше символов войдет в сочетания, тем меньше будут аппаратурные затраты и выше процент использования оборудования (памятей) платы. Уровень 4 структурной схемы АМП, как правило, реализуется на внутренних СОЗУ ПЛИС, а их объем сравнительно невелик: 18–36 Кбит с доступным адресным пространством 14–15 разрядов. На третьем уровне дешифрации с помощью специальных алгоритмических и технических решений восьмибитный код символа можно сократить до 6 бит. Следовательно, для трех символов, включенных в сочетание, необходима адресация 18-ти разрядов. Выполним расчет необходимого оборудования для сочетания символов С2–С4 в словах ЛКТ варианта *A* в соответствии с рис. 5 и табл. 1.

Ш а г 1. В качестве платы ПЛИС выбрана плата *HTG – V6HXT – X*16*PCIE –* 565 фирмы *HiTech Global* (*www.hitechglobal.com*/*boards*/*allboards.htm*), на которой установлена ПЛИС серии *Virtex* 6, имеющая в своем составе 912 СОЗУ разрядностью 32*K*×1 бит каждое, и внешняя память до 8 Гига 16-тиразрядных слов.

Ш а г 2. Количество сочетаний символов равно 26450 (см. табл. 1, сочетания С2–С4), следовательно, разрядность данных *RAM SS*_1 равна 15.

Ш а г 3. На один бит дешифрации требуется 8 СОЗУ (недостающие три разряда до 18 бит, необходимых для адресации сочетания из трех символов), а их общее количество для *RAM SS*_1 равно 8×15 = 120. Это самое большое количество СОЗУ для уровня 4–1 (рис. 2). Для сочетаний символов С5–С7 и С8–С10 необходимо меньшее количество СОЗУ. Следовательно, для указанных трех сочетаний символов (обеспечивающих так называемую независимую адресацию символов слов) необходимо около трети из общего количества СОЗУ.

Архитектурно-структурная организация АС морфологического анализа для ЛКТ варианта *A* представлена на рис. 5.

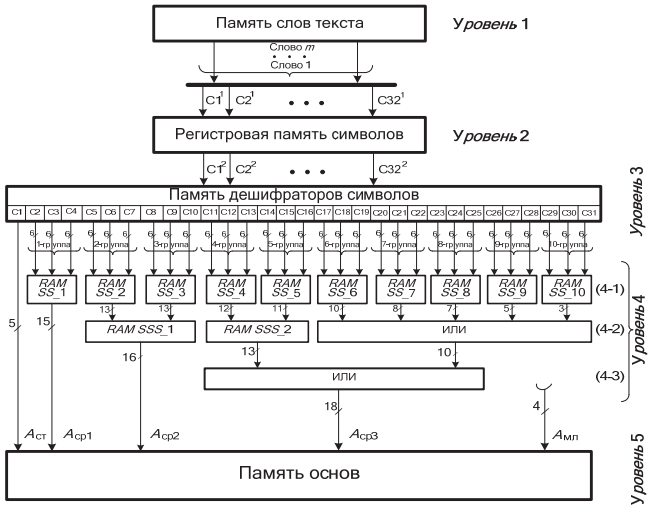

Рис. 5. Архитектурно-структурная организация АС МА для ЛКТ варианта *A*

Структуры вариантов B и C

Как видно из табл. 2 и 3 статистических характеристик ЛКТ вариантов *B* и *C*, структурные схемы АМП для указанных вариантов мало чем отличаются друг от друга. По сути, для варианта *C* требуется меньшее количество СОЗУ, что автоматически снижает процент использования оборудования, установленного на плате. При этом нецелесообразно использовать платы ПЛИС более ранних версий, так как у них разрядность адресов СОЗУ меньше, что повлечет необходимость увеличения количества таких СОЗУ, увеличение длительности синхросигналов процессора, да и объем внешней памяти *RAM* может быть критическим. Исследования показали, что разрабатывать АМП с внешней *RAM*, объем которой меньше (32–64)*M*×16 бит, нецелесообразно, так как при этом резко сокращается эффективность перевода операторов программного уровня на микропрограммный.

Структурная схема АМП вариантов *B* и *C* представлена на рис. 6.

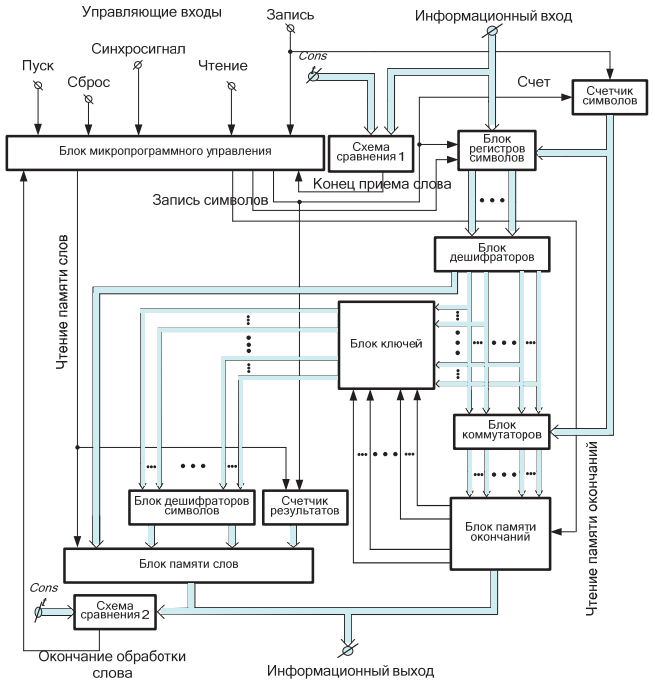

Рис. 6. Структурная схема АМП вариантов *B* и *C*

Работа АМП начинается с прихода на вход *Сброс* блока микропрограммного управления (БМУ) сигнала *Сброс* = 1, который инициирует в нем внутренний управляющий сигнал *Сброс*. Этот сигнал устанавливает в нулевое состояние блок регистров символов, регистр микрокоманд и счетчики символов и адресов результата. Затем АМП переходит в режим ожидания сигнала *Пуск* = 1. С его приходом АМП ожидает первый символ входного слова (*Запись* = 1). По его приходу на информационные входы блока регистров символов и первой схемы сравнения подается восьмибитный код первого символа (рассматривается байтовое кодирование символов, например *Win* 1251), и БМУ выдает сигнал *ЗпС* = 1. Номер символа записи формируется счетчиком символов, выходы которого управляют дешифратором. Выходы последнего – это управляющие сигналы записи в соответствующий регистр символа.

Восьмибитный код первого символа с выхода РгС1 дешифрируется в дешифраторе, с выхода которого пятиразрядный код формирует старшие адреса памяти слов. При этом объем последней памяти разбивается на 32 сегмента.

Аналогично в блок регистров символов записываются все символы входного слова. При этом по приходу очередного символа в схеме сравнения выполняется сравнение «код входного символа тождествен коду символа окончания передачи символов входного слова» (это может быть, например, код 09*H*, означающий *Пробел*), который постоянно находится на втором информационном входе первой схемы сравнения. По приходу кода символа окончания передачи символов входного слова на выходе первой схемы сравнения устанавливается сигнал «1», поступающий на соответствующий управляющий вход блока микропрограммного управления (БМУ).

Далее БМУ переходит к интерпретации алгоритма анализа символов входного слова (рис. 7). Сначала анализируются символы, возможно, принадлежащие окончанию входного слова. При этом их группировка важна для первых информационных входов блока ключей и не учитывается для информационных входов блока коммутаторов.

Рассмотрим алгоритм анализа окончания. При этом в счетчике символов будет записан код «01*H*», поступающий на вход мультиплексоров, на выходы которых будут переданы коды 0, 0, …, С*n*, передаваемые на адресные входы памяти окончаний.

В ячейке памяти окончаний с адресом 0, 0, …, С*n* записано:

- если символ С*n* не является окончанием и словом без основы, то на других информационных выходах памяти окончаний будет код *NOP* (нет операции), а на первых – код 0, 0, …, 0, т.е. на выходах блока ключей все символы С2, … …, С*n* (символы всех *q* групп) будут заблокированы. Блокирование символа означает, что в соответствующих разрядах выходов блока ключей, выходах блока дешифраторов сочетаний символов и средних адресов памяти основ будут коды 0, 0, …, 0;
- если символ С*n* – окончание и слово без основы, то будут выбраны адреса соответствующих ячеек памяти окончаний и памяти основ, в которых сохраняются результаты для окончания и слова С*n* соответственно.

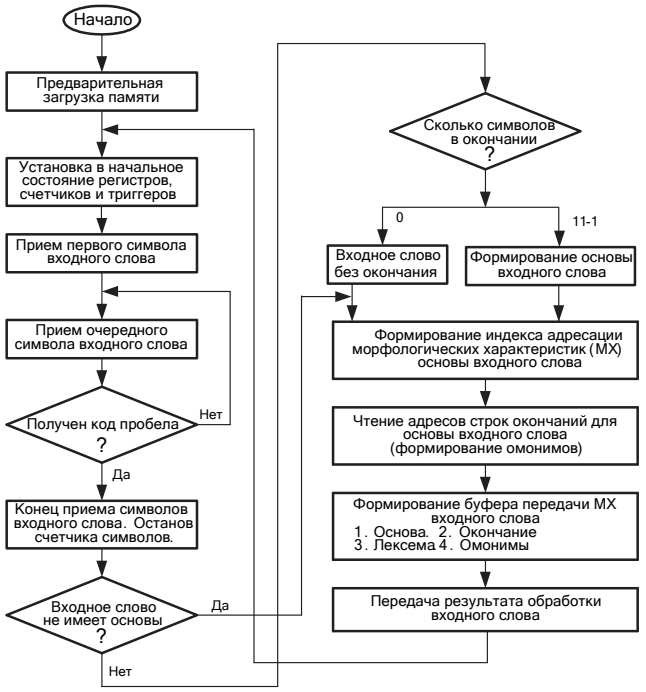

Рис. 7. Алгоритм работы АМП

Если входное слово состоит из двух букв (символов) С1 и С2 (они же при анализе окончаний интерпретируются как Сn – 1 и Сn), то блоком ключей символ С2 не будет заблокирован, и его дешифрированный код через блок дешифраторов сочетаний символов поступит на средние адреса памяти основ. На выходах блока коммутаторов будут присутствовать коды 0, …, Сn – 1, Сn, и в памяти окончаний будет выбран адрес результата анализа возможного окончания Сn – 1, Сn. На соответствующем выходе вторых информационных выходов памяти окончаний будет считан код результата. При этом управляющий выход БМУ *Чтение памяти окончаний* устанавливается в *единицу*.

Аналогично выполняется анализ для произвольной цепочки символов С1, …, Сn.

После анализа окончания и основы входного слова БМУ переходит к интерпретации микропрограммы выдачи результата анализа. Сначала на информационный выход АМП передается результат анализа основы входного слова. При этом управляющий выход БМУ *Чтение памяти слов* устанавливается в *единицу*, что обеспечивает чтение памяти основ и выполнение счета в счетчике адресов результата или выбор последовательных (по «+1») ячеек результата. Количество ячеек, в которых сохраняется результат, – это переменная величина и зависит от конкретной основы. Конечные ячейки каждого такого результата содержат коды, например 0D0AH, что означает конец передачи результата проанализированной основы. При этом информационные выходы памяти слов подключены к первому информационному входу второй схемы сравнения, выход которой, установленный в *единицу*, поступает на соответствующий управляющий вход БМУ и сигнализирует об окончании передачи результата основы.

Затем на информационный выход АМП передается код ячейки, содержащей результат анализа окончания (при этом управляющий сигнал БМУ *Чтение памяти окончаний* = 1 активен).

При каждой передаче слова результата на информационный выход АМП в БМУ анализируется управляющий вход *Чтение* = 1, сигнализирующий об окончании передачи очередного слова результата.

При завершении передачи на информационный выход АМП кодов всех ячеек результата БМУ на своем управляющем выходе устанавливает внутренний сигнал *Сброс* = 1, который устанавливает в *ноль* соответствующие регистры и счетчики, а алгоритм работы АМП переходит в режим ожидания приема очередного слова для анализа.

На описанную структуру АМП получен патент на полезную модель [4].

На рис. 8 показана диаграмма зависимости времени работы АМП (количества тактов обработки) от длины анализируемого слова (количества символов в слове).

Оценки сложности структурной реализации аппаратных морфологических процессоров

Ранее было рассмотрено проектирование АМП первого типа (с использованием ПЛИС-технологии) для трех вариантов лингвистических корпусов текстов, приведена производительность такого АМП в сравнении с программным способом реализации морфологического анализа для общеупотребительной лексики украинского языка.

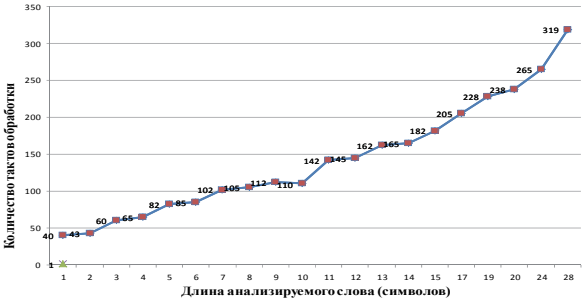

Рис. 8. Диаграмма зависимости времени работы АМП от длины слова

В табл. 4 приведены показатели моделирования АМП.

Таблица 4

Сравнительный анализ показателей морфологической обработки для программной и аппаратной реализаций АМП			
Вид реализации	Средняя длина слова	Время обработки (мкс)	Увеличение производительности для аппаратной реализации (раз)
Программный	9	937	252
Аппаратный	9	3,72	

Представляется целесообразным рассмотреть обобщенную архитектуру АМП второго типа (специальной разработки) и сравнить затраты оборудования и производительность для первого и второго типов реализации АМП (рис. 9). Для этого приняты следующие соглашения.

• Исследования показали, что схемы управления памятью АМП занимают сравнительно небольшую часть оборудования от его общего объема, и поэтому их можно не учитывать.

• Память слов текста занимает 2^{14} ячеек, что приблизительно равно среднему по объему научно-техническому тексту.

• Регистровая память символов и память дешифраторов символов рассчитаны на максимальную длину слов общеупотребительной лексики украинского языка (32 символа).

• При дешифрации символов учитывается восьмибитовый код, а не шестибитовый, как рассматривалось в АМП первого типа, что позволяет обрабатывать тексты на украинском, русском и английском языках, а также учитывать ряд специальных символов.

• Сравнение выполнено для затрат оборудования на реализацию АМП для ЛКТ варианта А в соответствии с архитектурно-структурной организацией, представленной на рис. 5.

• Как видно из диаграммы (рис. 3), количество слов длиной 14 символов и больше резко сокращается в сравнении с количеством слов меньшей длины. Поэтому можно принять ряд ограничений, наиболее существенным из которых будет «сборка по ИЛИ» выходов памяти сочетаний символов *SS*_5 – *SS*_10.

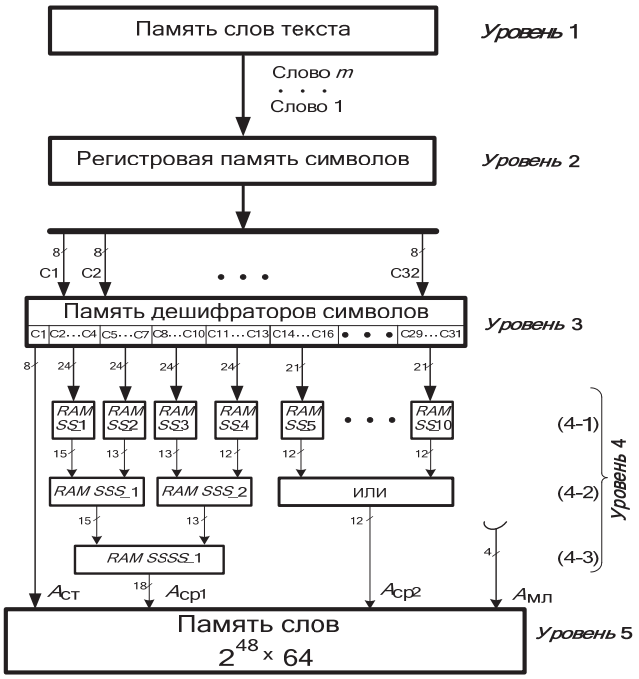

Рис. 9. Обобщенная архитектура АМП второго типа

Результаты сравнительного анализа затрат оборудования АМП первого и второго типов для морфологического анализа слов общеупотребительной лексики украинского языка представлены в табл. 5.

Из таблицы видно, что затраты памяти для АМП второго типа существенно ниже в сравнении с затратами памяти для АМП первого типа при одновременном повышении быстродействия на порядок и расширении функциональных возможностей. Это связано с проблемно-ориентированной структурной организацией АМП и выбором для каждого уровня архитектуры требуемых по объему и разрядности чипов памяти. При этом, как указывалось, сложность разработки АМП второго типа существенно выше.

Заключение. Анализ особенностей компьютерной обработки ЛКТ сверхбольших объемов показал, что для приложений, работающих в ре-

Таблица 5

Тип реализации	Затраты памяти (бит) по уровням иерархии $m = 1{,}6\bullet10^4$						Всего по уровням 1–4	Уровень 5
	Уровень 1	Уровень 2	Уровень 3	Уровень 4				
				4–1	4–2	4–3		
ПЛИС-технология	$4{,}2\bullet10^6$	$2{,}6\bullet10^2$	$4{,}7\bullet10^4$	$2{,}5\bullet10^7$	$1{,}2\bullet10^9$	–	$1{,}23\bullet10^9$	$2{,}9\bullet10^{17}$
Специальная разработка	$4{,}2\bullet10^6$	$2{,}6\bullet10^2$	$5{,}4\bullet10^4$	$1{,}1\bullet10^9$	$4{,}5\bullet10^9$	$4{,}8\bullet10^9$	$10{,}41\bullet10^9$	$2{,}8\bullet10^{14}$
Производительность АМП в сравнении с программным способом реализации морфологического анализа (раз)								
На базе ПЛИС							$2{,}5\bullet10^2$	
Специализированное оборудование							$2{,}6\bullet10^3$	

альном режиме времени, программной реализации лингвистического (и особенно морфологического) анализа недостаточно, так как часть информации может быть не обработана. Поэтому задача построения аппаратных лингвистических процессоров актуальна, и ее решение позволит: во-первых, сократить сроки предоставления пользователю оперативной информации (без потери части информации и снижения ее актуальности) для принятия решений; во-вторых, качественно повысить уровень лингвистических исследований с учетом большего количества параметров обработки.

Рассмотренные особенности архитектурно-структурной организации аппаратных морфологических процессоров для обработки ЛКТ разных объемов позволили выделить их лингвистические и статистические характеристики (основные – количество употребляемых словоформ K, их средняя длина $L_{ср}$ и количество сочетаний символов), непосредственно влияющие на количественные и качественные показатели архитектуры и структуры как АМП, так и лингвистической системы в целом. Статистические исследования выполнены на ЛКТ объемом 1 Гб.

Разработана архитектурно-структурная организация АС, реализующих этап МА для трех вариантов ЛКТ: общеупотребительной лексики украинского языка, онтологического инжиниринга и онтолого-управляемых информационных систем общего назначения. Для обработки указанных ЛКТ имеется три варианта структуры АМП, один из которых смоделирован в САПР ПЛИС *ISE Foundation* фирмы *Xilinx*. Сформулирована задача квазиоптимального синтеза структуры АМП на основе метода Парето.

Сравнительный анализ показателей морфологической обработки для программной и аппаратной реализаций МА показал, что повышение производительности для АС, выполненных по ПЛИС-технологии, составило два порядка, а для АС структурно-ориентированной разработки – три. Преимуществами АМП второго типа в сравнении с АМП первого типа есть снижение на три порядка объема памяти, повышение на порядок быстродействия и расширение функциональных возможностей.

1. *Палагин А.В.*, *Опанасенко В.Н.* Реконфигурируемые вычислительные системы. – К.: Просвіта, 2006. – 280 с.
2. *Палагин А.В.*, *Крывый С.Л.*, *Петренко Н.Г.* Онтологические методы и средства обработки предметных знаний. – Луганск: Изд-во ВНУ им. В. Даля, 2012. – 324 с.
3. *Палагин А.В.*, *Петренко Н.Г.* Методологические основы разработки лингвистического процессора для обработки ЛКТ сверхбольших объемов // УСиМ. – 2014. – № 2. – С. 44–57.
4. **Пат. № 104225**. Пристрій для морфологічного аналізу природномовних текстів / О.В. Палагін, М.Г. Петренко, В.Ю. Величко та ін. – Опубл. 10.01.2014, Бюл. № 1.

Поступила 17.02.2014

Тел. для справок: +38 044 526-3348 (Киев)

●